SIEMENS河南省开封市西门子变频器、驱动、PLC（授权）一级代理商——西门子华中总代理

产品名称	SIEMENS河南省开封市西门子变频器、驱动、PLC（授权）一级代理商——西门子华中总代理
公司名称	广东湘恒智能科技有限公司
价格	.00/件
规格参数	西门子总代理:PLC 西门子一级代:驱动 西门子代理商:伺服电机
公司地址	惠州大亚湾澳头石化大道中480号太东天地花园2栋二单元9层01号房
联系电话	15915421161 15903418770

产品详情

在 WinCC 项目运行时，在画面中捕捉到鼠标的坐标值。多年的热线支持养成的习惯总是会驱使我考虑一下用户要这个坐标值干嘛用呢？当然，问客户是最直接的，得到最多的回答就是希望获取这个坐标的目的在于：当在 WinCC 画面中点击设备图标时弹出设备的子画面，希望弹出子画面的坐标根据鼠标的坐标来自动确定,而不是在脚本中写为固定的常数。当然答案也不全是这个，也有比较特殊用途的。

其实为解决客户自动确定弹出子画面窗口坐标位置的这个问题无需大费周章，WinCC 的鼠标事件中其实就已经包含了鼠标坐标值。当然也并不是所有鼠标事件中都包含，首先来看一下 WinCC 按钮中都有哪些鼠标事件：

单击鼠标

按左键

释放左键

按右键

释放左键

那哪些事件能够简单的获取到鼠标坐标值呢？

很简单，只要随意打开一个按钮事件的动作脚本编辑器就能知道。

单击鼠标 C 动作：

按左键 C 动作：

按右键 VBS 动作：

细心的同学应该从截图中就已经找到答案了。除了单击鼠标动作事件中没有提供鼠标坐标值，其它动作事件中都已经默认提供了鼠标坐标值，也就是截图红框中的 x 和 y 。

为什么唯独单击鼠标事件没有提供鼠标坐标值呢？这是因为单击鼠标指的是鼠标按下并释放的一个完整过程，当鼠标指针在一个可操作对象上按下鼠标左键时，这个过程并未完成，此时如果想放弃操作只需要在按住左键不释放而将鼠标移动至可操作对象范围之外再释放，相当于就放弃了此次单击操作。所以 WinCC 只提供了鼠标按下以及释放时的鼠标坐标值就已经能完全满足需要了。

清楚了这一点，其实接下来要实现弹出窗口自动确定坐标的功能也就很容易实现了。首先简单测试一下如何通过 C 脚本以及 VB 脚本来获取并输出鼠标的 x 和 y 坐标值，然后再应用到实际需求中即可。

首先在画面中添加两个按钮以及一个应用程序窗口（窗口内容：全局脚本；模板：GSC Diagnostics）。

1.鼠标按左键时 C 脚本获取并输出 x 和 y 坐标值：

在按钮“ C 获取 x,y 坐标 ”的“ 按左键 ”事件中编写脚本

```
void OnLButtonDown(
char* lpszPictureName, char* lpszObjectName, char* lpszPropertyName, UINT nFlags, int x, int y)
{
printf("C 获取的 x y 坐标：X: %d Y: %d\r\n", x, y);
}
```

2.鼠标按左键时 VB 脚本获取并输出 x 和 y 坐标值：

在按钮“ VBS 获取 x,y 坐标 ”的“ 按左键 ”事件中编写脚本

```
Sub OnLButtonDown(ByVal Item, ByVal Flags, ByVal x, ByVal y)
```

HMIRuntime.Trace "VBS 获取的 x y 坐标：" & " X: " & x & " Y: " & y & vbCrlf

End Sub

激活 WinCC 运行系统后，在画面中鼠标分别按下两个按钮后，应用程序窗口中将会输出鼠标按下时的 x，y 坐标值。

接下来就可以应用到实际需求中了。将刚才的两个按钮当作设备图标，当按下两个按钮时自动根据鼠标位置弹出子画面窗口（以 VBS 为例）。

1.添加并编辑一个子画面（DeviceDetail.pdl）

2.主画面中添加一个画面窗口（对象名称：SubPic）

3.给两个按钮编重新编写 VB 脚本

激活运行后，分别按下两个按钮时，弹出窗口则会自动根据鼠标按下时的坐标位置弹出

以上方法都是由鼠标动作事件来获取鼠标坐标值，如果希望在鼠标移动而不执行鼠标动作事件时也实时获取鼠标位置是否可行呢？答案是肯定的，通过强大的 C 脚本也是可以实现的，方法如下：

在 C 全局动作中添加一个全局动作，并将触发器设置为周期 250 ms

当项目激活运行后，在应用程序窗口中就可以看到实时变化的鼠标坐标值。

通过这种方法也可以做出一些特殊的效果，例如导航菜单根据鼠标位置的自动弹出及隐藏。

重要提示

这种做法虽然可以实时获取到鼠标坐标值，但是毕竟是通过高频的执行 C 脚本来实现的。熟悉 WinCC 的工程师都知道，由于 WinCC 的脚本都是队列执行的，这种做法实际会很占用 WinCC 脚本运行资源。因此，在这里只是抛砖引玉，实际使用中千万要慎用！